On the decision problem for formulas in which all disjunctions are binary.

By

Stål O. Aanderaa

Institute of Mathematics
University of Oslo
1970 *)

*) The first draft of this paper was written when the author was guest-investigator at Rochefeller University, New York, March-May 1970.

Abstract. Let Z_1 be the class of closed formulas of the form $\exists a \forall y\ Kay \ \& \ \forall x \exists u \forall y\ Mxuy$ where Kay and $Mxuy$ are conjunctions of binary disjunctions of signed atomic formulas of the form $F\alpha\beta$ or $\neg F\alpha\beta$ where F is a binary predicate symbol, and α and β are one of the variables a, x, u and y. We prove in our paper that there is no recursive set which separates the non-satisfiable formulas in Z_1 from those satisfiable in a finite domain.

§ 1. Introduction. In order to state the result of this paper, it is convenient first to introduce some definitions.

Definition 1. For any class of formulas X let $N(X)$, $I(X)$ and $F(X)$ be the subclasses of X which contain all formulas in X which have respectively, no model, only infinite models, finite models.

Note that $N(X)$ and $F(X)$ are r.e. (recursive enumerable) if X is r.e.

Definition 2. A class of formulas X is a Trachtenbrot class if $N(X)$ and $F(X)$ are recursively inseparable.
(Two disjoint sets A_0 and A_1 are recursively inseparable if there exists no recursive set B such that $A_0 \subseteq B$ and $A_1 \cap B = \emptyset$.)

Note that if X is a Trachtenbrot class, then neither $N(X)$ nor $F(X)$ nor $I(X)$ are recursive.

Trachtenbrot (1953) proved that the class of all formulas in first order predicate calculus is a Trachtenbrot class.

We shall here deal with formulas which are in prenex normal form or which are a conjunction of formulas in prenex normal form.

<u>Definition 3</u>. Let $Q_1, Q_2, \ldots, Q_n$ $(n = 1,2,\ldots)$ be a sequence of strings of quantifiers. Then a formula S is a Q_1 & Q_2 &...& Q_n-formula iff S is a closed formula in first order predicate calculus of the form Q_1M_1 & Q_2M_2 &...& Q_nM_m where M_i $(i = 1,2,\ldots n)$ is quantifier-free and contains neither the equality sign nor function symbols. The Q_1 & Q_2 &...& Q_n - class is the class of all Q_1 & Q_2 &...& Q_n- formulas. If X is a class of formulas then a Q_1 & Q_2 &...& $Q_n \cap X$ - formula is a formula which is both a Q_1 & Q_2 &...& Q_n - formula and a formula in X . The Q_1 & Q_2 &...& $Q_n \cap X$ - class is the intersection of the classes Q_1 & Q_2 &...& Q_n and X .

Hao Wang (1962) has proved that both the $\forall\exists\forall$ class and the $\forall\forall\forall\exists$ class are Trachtenbrot classes. But $I(\exists\exists\ldots\exists\forall\forall\ldots\forall$-class) and $I(\exists\exists\ldots\exists\forall\forall\exists\exists\ldots\exists$-class) are empty classes. Hence the classes $N(\exists\exists\ldots\exists\forall\forall\ldots\forall$-class), $F(\exists\exists\ldots\exists\forall\forall\ldots\forall$-class), $N(\exists\exists\ldots\exists\forall\forall\exists\exists\ldots\exists$-class) and $F(\exists\exists\ldots\exists\forall\forall\exists\exists\ldots\exists$-class) are all recursive. This shows that the $\exists\exists\ldots\exists\forall\forall\ldots\forall$-classes and the $\exists\exists\ldots\exists\forall\forall\exists\exists\ldots\exists$-classes are not Trachtenbrot classes. Hence the problem of deciding whether a prefix class is a Trachtenbrot class is solved. These problems are in fact also solved for such classes as Q_1 & Q_2 &...& Q_n-classes.

If we also put some restrictions on the matrix, then new cases occur. Some of these cases have been solved. We may classify the formulas according to the atomic subformulas. See Dreben, Kahn, Wang 1962 and Wang 1962, and Aanderaa 1966.

Melvin R. Krom and S.Ju Maslov have studied formulas in which the matrices consist of conjunctions of binary disjunctions. See Krom 1962, 1967a, 1967b, 1968, 1970 and Maslov 1964. The aim of this paper is also to investigate such formulas. It is therefore convenient to introduce the following definition.

Definition 4. Let A be a formula in first-order predicate calculus and let A' be the result of deleting the quantifiers in A. Then A is a Krom formula iff A' consists of a conjunction

$$C_1 \;\&\; C_2 \;\&\ldots\&\; C_m \tag{2}$$

of binary disjunctions $C_i = D_{1i} \vee D_{2i}$ of signed atomic formulas D_{1i}, D_{2i}, $i = 1,2,\ldots,m$. Each term C_i in (2) is called a conjunct of A'. The class of Krom formulas is denoted by Kr.

Note that to each Krom formula A, there corresponds a Krom formula B in prenex normal form such that $\vdash A \equiv B$.

The main theorem of the first part of this paper is:

Theorem 1. The $\exists\forall \;\&\; \forall\exists\forall \cap$ Kr-class is a Trachtenbrot class.

From theorem 1 follows immediate the following corollaries.

Corollary 1. The decision problem for the $\exists\forall \;\&\; \forall\exists\forall \cap$ Kr-class is recursively unsolvable.

Corollary 2. The $\exists\forall\exists\forall \cap$ Kr-class is a Trachtenbrot class.

Corollary 3. The $\forall\exists\exists\forall \cap$ Kr-class is a Trachtenbrot class.

Corollary 4. The decision problem for the $\exists\forall\exists\forall \cap$ Kr-class is recursively unsolvable.

Corollary 5. The decision problem for the $\forall\exists\exists\forall \cap$ Kr-class is recursively unsolvable.

Krom 1970 has proved a weaker form of corollary 5. He proved that the decision problem for the class of Krom formulas in prenex normal form with a prefix of the form $\forall\exists\exists\ldots\exists\forall$ is recursively undecidable.

We shall also prove the following theorem in the first part of this paper:

<u>Theorem 2</u>. The classes $\exists\forall \;\&\; \forall\exists\forall \cap \mathrm{Kr}$, $\exists\forall\exists\forall \cap \mathrm{Kr}$ and $\forall\exists\exists\forall \cap \mathrm{Kr}$ are reduction classes.

It turns out, however, that the classes $N(\forall\exists\forall \cap \mathrm{Kr}\text{-class})$ and $F(\forall\exists\forall \cap \mathrm{Kr}\text{-class})$ are recursive. See § 4 in this paper.

Maslov 1964 has proved that the class $N(\exists\exists\ldots\exists\forall\forall\ldots\forall\exists\exists\ldots\exists \cap \mathrm{Kr}\text{-class})$ is recursive.

We shall give the proofs of theorems 1 and 2 in detail; and our intention is that the proofs should be elementary. We shall reduce an output problem for registermachines to the problem of deciding whether $\exists\forall \;\&\; \forall\exists\forall$-formulas are consistent or have a finite model.

We shall only define registermachines and state the result we need from the theory of registermachines. Two-registermachines are called 2-type non-writing machines in Minsky 1961. n-register-machines are called program machines in Minsky 1967, pp. 199-215, and two-register machines are studied on pp. 255-258. Register-machines are also some times called countermachines. By using an appropriate coding, 2-registermachines may be used to define recursive functions. See for instance Fischer 1966, Minsky 1962, Minsky 1967, Shepherdson 1965, or Shepherdson and Sturgis 1963.

We shall first establish a lemma about registermashines and recursively unseparable sets in § 2. We shall prove the theorems 1 and 2 in § 3. Finally, we shall state some further new results in § 4. But since these results seems to be of less importance we shall in § 4 only sketch the proofs.

§ 2. The n-registermachine. An n-register machine R_n consists of n registers (or also called counters) $T_1, T_2, \ldots, T_n$, capable of storing arbitrary large natural numbers, R_n is programmed by a numbered sequence $I_1, I_2, \ldots, I_r$ of instructions. An instantaneous description (abbreviated ID) of R is denoted by

$$(i, x_1, x_2, \ldots, x_n)$$

(where $1 \leq i \leq r$ and $x_1, x_2, \ldots, x_n \geq 0$) and describes R_n ready to execute instruction I_i, with registers $T_1, T_2, \ldots, T_n$ containing $x_1, x_2, \ldots, x_n$, respectively. The instructions are all chosen from the instruction repertoire,

$$\{H_o, H_1, P(h), DJ(h,j) \ (h = 1,2,\ldots,n \quad j = 1,2,\ldots,r)\}.$$

Here

H_o means: halt and output zero.

H_1 means: halt and output 1.

$P(h)$ means: add 1 to the contents of register number h. Go on to next instruction.

$DJ(h,j)$ means: If contents of register number h is not zero, decrease it by 1 and jump to instruction number j. If contents of register number h is zero, go on to next instruction.

A register machine R is defined when its program

(3) $\quad I_1, I_2, I_3, \ldots, I_r$

is defined. In order to deal with computations we shall introduce relations $\vdash_R$ and $\vdash_R^*$ between ID's (see table 1). We shall often write $\vdash$ and $\vdash^*$ for $\vdash_R$ and $\vdash_R^*$ when no confusion results. From now on we shall deal mainly with 2-register machines.

We shall use i, j with or without subscript to denote numbers of the set $\{1,2,\dots,r\}$ and α and β with or without subscript to denote non-negative integers. Then $\vdash(i,\alpha,\beta)$ means that (i,α,β) is the initial ID. In § 3 $\vdash(i,\alpha,\beta)$ iff $(i,\alpha,\beta) = (1,0,0)$.
$(i_1,\alpha_1,\beta_1) \vdash_R (i_2,\alpha_2,\beta_2)$ means that the ID (i_2,α_2,β_2) follows immediately from (i_1,α_1,β_1) according to the program (3).
If $z = 0$ or $z = 1$ then $(i,\alpha,\beta) \vdash z$ means that (i,α,β) is a halting state with output z, 0 and 1 are called improper instantaneous descriptions, (proper instantaneous description are of the form (i,α,β) where i,α,β are non-negative integers and $1 \leq i \leq r$). Suppose that the relation $\vdash$ is defined, Then we define $\vdash^*$ as follows.

<u>Definition 5.</u> $b \vdash^*_R c$ means that either $b = c$ or there exist ID's $d_0, d_1, \dots, d_n$ $(n \geq 0)$ where $d_0 = b$ and $d_n = c$ and $d_k \vdash_R d_{k+1}$ $(k = 0,1,\dots,n-1)$. Moreover, $\vdash^*_R c$ iff there exists an a such that $a \vdash^*_R c$ and $\vdash_R a$. We shall say that c is an <u>immediate successor</u> of b iff $b \vdash_R c$. Moreover, c is a <u>successor</u> of b iff $b \vdash^*_R c$.

We shall in the next section give a precise definition of the relation $\vdash$ and at the same time associate a formula to each 2-register machine R.

<u>§ 3. Reduction to 2-register machines.</u>

To each 2-register machine R with program (3), we shall associate a first order language L_R and a formula S_R in L_R. To each instruction I_i we associate a binary predicate letter F_i. The intended interpretation of F_i is an interpretation over the non-negative integers such that $F_i\alpha\beta$ is true iff

$\vdash^{*}(i,\alpha,\beta)$.

We shall now define the relation $\vdash_R$ describing R's operation on its ID's . At the same time we shall define the Krom formula of the form

(4) $\exists a \forall y\ Kay\ \&\ \forall x\ \exists u\ \forall y\ Mxuy$

which correspond to R , by listing its binary disjunctions. Each binary disjunction C is a conjunct in Kay iff neither x nor u occur in C and C is a conjunct in Mxuy iff a does not occur in C .

Then $\vdash_R$ and (4) are defined according to the following table, which is constructed according to the numbered sequence of instruction (3).

Case	If I_i =	then the relation $\vdash$ is defined to satisfy	and the following binary disjuntion is added
0	$= I_1$	$\vdash(1,0,0)$	$F_i aa \vee F_2 aa$
1	$= P(1)$	$(i,\alpha,\beta)\vdash(i+1,\alpha+1,\beta)$	$\neg F_i xy \vee F_{i+1} uy$
2	$= P(2)$	$(i,\alpha,\beta)\vdash(i+1,\alpha,\beta+1)$	$\neg F_i yx \vee F_{i+1} yu$
3	$= D(1,j)$	$(i,\alpha+1,\beta)\vdash(j,\alpha,\beta)$	$\neg F_i uy \vee F_j xy$
4	$= D(1,j)$	$(i,0,\beta)\vdash(i+1,0,\beta)$	$\neg F_i ay \vee F_{i+1} ay$
5	$= D(2,j)$	$(i,\alpha,\beta+1)\vdash(j,\alpha,\beta)$	$\neg F_i yu \vee F_j yx$
6	$= D(2,j)$	$(i,\alpha,0)\vdash(i+1,\alpha,0)$	$\neg F_i ya \vee F_{i+1} ya$
7	$= H_0$	$(i,\alpha,\beta)\vdash 0$	$\neg F_i xy \vee \neg F_i xy$
8	$= H_1$	$(i,\alpha,\beta)\vdash 1$	$\neg F_i xy \vee F_i xy$

Table 1.

Note that in case 8 in table 1, the binary disjunction is a tautology. Hence we may in this case add no binary disjunction as well.

Definition 6. $b \vdash^{*} c$ means that ID's $d_o, d_1, \ldots, d_n$ $(n \geq 0)$ exist where $d_o = b$, $d_n = c$ and $d_o \vdash d_1 \vdash \ldots \vdash d_n$. Moreover, $\vdash^{*} c$ iff $(1,0,0) \vdash^{*} c$. We shall say that c is an immediate successor of b iff $b \vdash c$. Moreover, c is a successor of b iff $b \vdash^{*} c$.

Definition 7. A finite computation (or a converging computation) is a finite sequence of ID's $d_1, d_2, \ldots, d_{m-1}, d_m$ such that $d_i \vdash d_{i+1}$ $(i = 1,2,\ldots,m-1)$ and d_{m-1} is terminal and d_m is an improper ID.

An infinite computation (or a diverging computation) is an infinite sequence of ID's $d_1, d_2, d_3, \ldots$ such that $d_i \vdash d_{i+1}$ $(i = 1,2,3,\ldots)$

We shall in § 4 consider computations where d_1 is $(1,0,0)$.

Example 1. Consider the following 2-register machine R_1 and its corresponding formulas. $\exists a \forall y K_1 ay \;\&\; \forall x \exists u \forall y M_1 xuy$

I_1	DJ(1,6)	$F_1aa \vee F_1aa$, $\neg F_1uy \vee F_6xy$, $\neg F_1ay \vee F_2ay$
I_2	P(1)	$\neg F_2xy \vee F_3uy$
I_3	DJ(2,1)	$\neg F_3yu \vee F_1yx$, $\neg F_3ya \vee F_4ya$
I_4	H_o	$\neg F_4xy \vee \neg F_4xy$
I_5	P(2)	$\neg F_5yx \vee F_6yu$
I_6	H_1	$\neg F_6xy \vee F_6xy$

Hence

K_1ay is

$(F_1aa \vee F_1aa) \;\&\; (\neg F_1ay \vee F_2ay) \;\&\; (\neg F_3ya \vee F_4ya)$

and M_1xyu is

$(\neg F_1uy \vee F_6xy) \;\&\; (\neg F_2xy \vee F_3uy) \;\&\; (\neg F_3yu \vee F_1yx)$

$(\neg F_4xy \vee \neg F_4xy) \;\&\; (\neg F_5yx \vee F_6yu) \;\&\; (\neg F_6xy \vee F_6xy)$.

The following is the computation from empty regisers in example 1 : (1,0,0), (2,0,0), (3,1,0), (4,1,0), 0 .

Consider the following examples.

	Example 2	Example 3
I_1	DJ(1,6)	P(1)
I_2	P(1)	P(1)
I_3	P(2)	DJ(1,1)
I_4	DJ(2,1)	H_o
I_5	H_o	H_1
I_6	H_1	

The computation according to example 2 from empty registers is:

(1,0,0), (2,0,0), (3,1,0), (4,1,1),
(1,1,0), (6,0,0), 1 .

The computation from empty registers according to example 3 is infinite, and the first seven ID's are

(1,0,0), (2,1,0), (3,2,0), (1,1,0), (2,2,0),
(3,3,0), (1,2,0), (2,3,0), ...

Let $d_1, d_2, \ldots, d_m$ be a finite computation. The <u>output</u> of the computation from d_1 is then d_m . If a computation diverges, then the output is not defined.

Consider now computation where the initial ID is $(1,2^i,0)$ for some i . Then each register machine R as defined above defines a partial recursive function ψ such that range $\psi \subseteq \{0,1\}$

and such that

(5) $$(1,2^i,0) \vdash_R^* 0 \iff \psi(i) = 0$$

(6) $$(1,2^i,1) \vdash_R^* 1 \iff \psi(i) = 1 .$$

Moreover, given a partial recursive function ψ such that range $\psi \subseteq \{0,1\}$ there exists a register machine R such that (5) and (6) are satisfied. (See Minsky 1967 p.257 or Hopcroft and Ullman 1969, p.100 or Fischer 1966, p.377).

The following lemma is then easily proved by standard methods in recursion theory. (See Rogers 1961, p.94).

Lemma 1. Let N^+, I^+ and F^+ be the set of 2-register machines such that the output of the computation from empty registers are 0, not defined and 1, respectively. Then N^+ and F^+ are recursively inseparable.

Proof. Suppose that there exist a recursive set A such that $F^+ \subseteq A$ and $N^+ \cap A = \emptyset$. (We shall prove lemma 1 by obtaining a contradiction from this assumption.) Then there exists recursive function f such that range $f = \{0,1\}$ and such that

$$R_x \in A \implies f(x) = 0 .$$

$$R_x \notin A \implies f(x) = 1 .$$

Here R_x is the 2-register machine with gödel number x. Hence

$$R_x \in F^+ \implies f(x) = 0$$

$$R_x \in N^+ \implies f(x) = 1 .$$

There exists a recursive function h such that

$$(1,2^x,0) \vdash_{R_x}^* 1 \iff (1,0,0) \vdash_{R_{h(x)}}^* 1$$

and

$$(1,2^x,0) \vdash^{*}_{R_x} 0 \iff (1,0,0) \vdash^{*}_{R_{h(x)}} 0$$

Let $g(x) = f(h(x))$. Then g is a recursive function and range $g = \{0,1\}$. Then

(7) $\quad (1,2^x,0) \vdash^{*}_{R_x} 1 \implies g(x) = 0$

(8) $\quad (1,2^x,1) \vdash^{*}_{R_x} 0 \implies g(x) = 1$.

Choose z such that

(9) $\quad (1,2^x,0) \vdash^{*}_{R_z} 1 \iff g(x) = 1$

(10) $\quad (1,2^x,0) \vdash^{*}_{R_z} 0 \iff g(x) = 0$

Substituting z for x in (7) and (8) we obtain a contradiction. This proves lemma 1.

<u>Lemma 2.</u> The effective mapping Π_1 of 2-register machines into the $\exists\forall$ & $\forall\exists\forall \cap Kr$-class of formulas defined by (4) and table 1, satisfies the following conditions.

(11) $\quad R \in N^+ \implies \Pi_1(R) \in N(\exists\forall \,\&\, \forall\exists\forall \cap Kr)$

(12) $\quad R \in F^+ \implies \Pi_1(R) \in F(\exists\forall \,\&\, \forall\exists\forall \cap Kr)$.

(Here N^+ and F^+ are defined as in lemma 1.).

<u>Proof.</u> We shall first prove (11).

Suppose that $R \in N^+$, and suppose that $\Pi(R)$ is consistent. We shall prove that this is a contradiction. Since $R \in N^+$ we have that $\vdash^{*}_{R} 0$ i.e. $(1,0,0) \vdash^{*}_{R} 0$.

The formula $\Pi_1(R)$ is of the form

(4) $\quad \exists a \forall y\, Kay \,\&\, \forall x \exists u \forall y\, Mxuy$.

We can now use a either modeltheoretic argument or a syntactical argument. Since (3) has a model there exist elements in $a_o, a_1, a_2, \ldots, a_n$ such that

(13) $\forall y Ka_o y$ & $\forall y Ma_o a_1 y$ & $\forall y Ma_1 a_2 y$ & ... & $\forall y Ma_{n-1} a_n y$

is true. Here we choose n larger than the maximum of the content of the registers in the computation

(14) $d_o, d_1, \ldots, d_t, \ldots, d_{m-1}, d_m$

where $d_o = (1,0,0)$ and $d_m = 0$.

Hence we have that

(15.β) $Ka_o a_\beta$ & $Ma_o a_1 a_\beta$ & $Ma_1 a_2 a_\beta$ & ... & $Ma_{n-1} a_n a_\beta$

for $\beta = 0,1,\ldots,n$.

We shall now prove that if $d_t = (j,\alpha,\beta)$ in (14) $(t = 0,1,2,\ldots,m-1)$, then $F_j a_\alpha a_\beta$ is true. The proof is by induction on t . If $t = 0$, then $d_t = d_o = (1,0,0)$, since $a_o = a$ we have according to case 0 in table 1 that $F_1 a_o a_o$ is true. Suppose that $d_{t-1} = (i,\alpha,\beta)$ and $d_t = (j,\gamma,\delta)$ and that $F_i a_\alpha a_\beta$ is true. We shall prove that $F_j a_\gamma a_\delta$ is true. We have 6 cases according to table 1. Suppose that I_i is the instruction P(1) (case 1). Then $d_t = (i+1,\alpha+1,\beta)$. We shall prove that $F_{i+1} a_{\alpha+1} a_\beta$. $Ma_\alpha a_{\alpha+1} a_\beta$ is true according to (15.β). According to case 1 $\neg F_i xy \vee F_{i+1} uy$ is a binary disjunction (conjunct) Mxuy . Hence $\neg F_i a_\alpha a_\beta \vee F_{i+1} a_{\alpha+1} a_\beta$ is a conjunct in $Ma_\gamma a_{\beta+1} a_\beta$. Hence $\neg F_i a_\alpha a_\beta \vee F_{i+1} a_{a+1} \beta_\beta$ is true. Moreover, $F_i a_\alpha a_\beta$ is true by induction hypothesis, since $d_t = (i,\alpha,\beta)$. Hence $F_{i+1} a_{\alpha+1} a_\beta$ is true. This completes the induction proof in the case $d_{t-1} = (i,\alpha,\beta)$ and I_i is P(1). The case I_i is P(2) (case 2) is proved in the same way. If I_i is D(1,j) then we have to distinguish between the case 3 where $d_{t-1} = (i,\alpha,\beta)$ and $\alpha > 0$ and the case 4 where $\alpha = 0$.

In case 3 we have that $\neg F_i a_\alpha a_\beta \vee F_j a_{\alpha-1} a_\beta$ is a conjunct of $Ma_{\alpha-1} a_\alpha a_\beta$. In case $\alpha = 0$ (case 4) we have that $\neg F_i a_o a_\beta \vee F_{i+1} a_o a_\beta$ is a conjunct of $Ka_o a_\beta$ in $(15.\beta)$. Otherwise the proof is as before. Hence we have in particular that if $d_{m-1} = (j,\alpha,\beta)$ then $F_j a_\alpha a_\beta$ is true, But I_j is H_o. Hence we have that $\neg F_j a_\alpha a_\beta \vee \neg F_j a_\alpha a_\beta$ is true since $\neg F_j a_\alpha a_\beta \vee \neg F_j a_\alpha a_\beta$ is a conjunct in $Ma_\alpha a_{\alpha+1} a_\beta$ which is true according to $(15.\beta)$. Then $F_j a_\alpha a_\beta$ is false, which is a contradiction. This completes the proof of (11).

Note that we can easily obtain a syntactical proof by proving that (4) implies the existential closure of (13) which in turn implies the existential closure of the conjunction

$$(15.0) \;\&\; (15.1) \;\&\; \dots \;\&\; (15.n) .$$

Then the argument is as before, except for replacing "true" by "provable".

To prove (12) let

(16) $$d_o, d_1, \dots, d_t, \dots, d_{m-1}, d_m$$

be a computation where $d_o = (1,0,0)$ and $d_m = 1$. Let n be larger than the maximal content of the registers in the computation (16). It is easy to verify that the formula $\Pi_1(R)$ of the form (4) is satisfiable in an infinite domain $\{a_o, a_1, a_2, \dots\}$ where $F_i a_\alpha a_\beta$ is true iff (i,α,β) occurs in (16).
To see this we choose $a = a_o$ in $\exists a \forall y Kay$ and if x in $\forall x \exists u \forall y Mxuy$ has the value a_α then we pick the value $a_{\alpha+1}$ for u. Since n was larger than the maximal content of the registers, we have that $F_i a_\alpha a_\beta$ is false if $\alpha \geq n+1$ or $\beta \geq n+1$. Hence we have that $F_i a_\alpha a_\beta \equiv F_i a_n a_\beta$ and $Fa_\beta a_\alpha \equiv Fa_\beta a_n$ if $\alpha \geq n$. Hence (4) is satisfiable in the domain $\{a_o, a_1, a_2, \dots, a_n\}$, where $F_i a_\alpha a_\beta$ is true iff (i,α,β) occurs in (16).

This proves (12) and the proof of lemma 2 is complete.

To prove theorem 1, suppose that theorem 1 is false. Then there exists a recursive set Y_1 which separates $N(Z_1)$ and $F(Z_1)$ *), i.e. $F(Z_1) \subseteq Y_1$ and $N(Z_1) \cap Y_1 = \emptyset$. Let Π_1 be the mapping mentioned in lemma 2. Let S be the set $\Pi_1^{-1}(Y_1)$ i.e.

(17) $\quad R \in S \iff \Pi_1(R) \in Y_1$.

Let N^+ and F^+ be as in lemma 1. Then $N^+ \cap S = \emptyset$ since suppose that $R \in N^+$, then $\Pi_1(R) \in N(Z_1)$ by (11). But $\Pi_1(R) \in N(Z_1)$ and $N(Z_1) \cap Y_1 = \emptyset$. Hence $\Pi_1(R) \notin Y_1$ and therefore $R \notin S$ by (17). Hence $N^+ \cap S = \emptyset$. Moreover $F^+ \subseteq S$, since suppose $R \subseteq F^+$. Then $\Pi(R) \in F(Z_1)$ by (12). Hence $\Pi(R) \in Y_1$ since $F(Z_1) \subseteq Y_1$, hence $R \in S$. This shows that $F^+ \subseteq S$. But S is recursive since the set Y_1 is recursive and the mapping Π_1 is recursive. Hence S is a recursive set which separates N^+ and F^+, which is impossible according to lemma 1. This proves theorem 1.

To prove theorem 2, we use the following familiar fact in logic: (see H. Wang 1962)

<u>Lemma 3</u>. There is an effective partial procedure by which, given a formula in first order perdicate calculus, we can test whether it has no model, a finite model, or only infinite models. The procedure terminates in the first two cases but does not terminate in the last case.

Hence, given a formula S in first order predicate calculus, we can effectively construct a 2-Register machine $R(S)$, which gives output 0 if S has no model, and output 1 if S has a finite model, and diverges otherwise. Then by lemma 2, $\Pi_1(R(S))$ is consistent iff S is consistent. Hence $\exists\forall \& \ \forall\exists\forall \cap Kr$ is a

*) Z_1 is the class of $\exists\forall \& \ \forall\exists\forall \cap Kr$-formulas

reduction class. This proves theorem 2 .

§ 4. Further results.

We shall state some further results which may be proved by refinement of the technique used so far. Since the result seems to be of less importance than the earlier results we shall only sketch the proof. First we shall here consider other computations started on empty registers. The initial values of the registers are called input. In the construction of the formulas, input may be taken care of by adding new monadic predicate letters G_i and H_i . Intuitively $G_i x$ means $x = i$ and $H_i x$ means $x = 2^i$. Binary disjunctions of the form $G_0 a \vee G_0 a$ and $\neg G_i x \vee G_{i+1} u$, $G_{i+1} u \vee \neg G_i x$, $\neg G_j x \vee H_i x$ and $G_j x \vee \neg H_i x$ where $j = 2^i$ will take care of the input.

As far as model theory is concerned, we shall partly follow Shoenfield with respect to notions and notations. See Shoenfield 1967 p. 14-23. Let $\mathcal{A}$ be a structure for a first order language L . $|\mathcal{A}|$ is the <u>universe</u> of $\mathcal{A}$ and the elements of $|\mathcal{A}|$ are called the <u>individuals</u> of $\mathcal{A}$. Then $L(\mathcal{A})$ is the first order language obtained from L by adding all the names of individuals of $\mathcal{A}$. If A is a closed formula in $L(\mathcal{A})$, let $\mathcal{A}(A) = T$ mean that A is true in $\mathcal{A}$. This is also often expressed by $\mathcal{A} \models A$ or $\models_{\mathcal{A}} A$. Let Γ^+ be the set of atomic formulas in $L(\mathcal{A})$ and let Γ^- be the set of negation of formulas in Γ^+ . Following Robinson 1963 p.24, we define <u>the positive diagram</u> $D^+(\mathcal{A})$ of $\mathcal{A}$ to be the set of formulas A in Γ^+ such that $\mathcal{A}(A) = T$, and <u>the negative diagram</u> $D^-(\mathcal{A})$ of $\mathcal{A}$ is the set of formulas A in Γ^- such that $\mathcal{A}(A) = T$. The <u>diagram</u> $D(\mathcal{A})$ of $\mathcal{A}$ is the set $D^+(\mathcal{A}) \cup D^-(\mathcal{A})$. If $\Gamma = \Gamma^+ \cup \Gamma^-$, note that $D^+(\mathcal{A})$, $D^-(\mathcal{A})$

and $D(\mathfrak{A})$ in the sense of Robinson 1963 p.24 correspond to $D_{\Gamma^+}(\mathfrak{A})$, $D_{\Gamma^-}(\mathfrak{A})$ and $D_{\Gamma}(\mathfrak{A})$, respectively, in the sense of Shoenfield 1967 p.74.

Definition 8.

Let Z be a class of formulas where each formula in Z is no more complex than

(18) $\exists a \forall x \exists u \forall y Maxuy$

or a conjunction of formulas of the form (18) or simpler. Here $Maxuy$ is quantifier-free. A Büchi model for the class Z is a model $\mathfrak{A}$ such that $|\mathfrak{A}| = \{0,1,2,\dots\}$ and such that for each part of the form (18), we have that $\mathfrak{A}(MOnn'm) = T$ where $n' = n+1$, for each number n and m.

The main theorem in this section is:

Theorem 3. Let A_0 and A_1 be two disjoint r.e. sets. Then there exist two sequences of Krom formulas $B_i(A_0,A_1)$ and $B_i'(A_0,A_1)$ of the forms

(19) $B_i = B_i(A_0,A_1) = \exists a \forall y N_1 ay \,\&\, \forall x \exists u \forall y (N_2 xuy \,\&\, M_i xu) \,\&\, \forall x (\neg P_i x \vee F_0 x)$

(20) $B_i' = B_i'(A_0,A_1) = \exists a_0 \exists a_1 \dots \exists a_j \forall y (N_1 a_0 y \,\&\, N_2 a_0 a_1 y \,\&\, N_2 a_1 a_2 y \,\&\dots\&\, N_2 a_{j-1} a_j y \,\&\, F_0 a_j) \,\&\, \forall x \exists u \forall y N_2 xuy$, $\quad (j = 2^i)$.

where $M_i xu$ is an initial segment of an infinite conjunction Mxu. Moreover, Mxu contains only monadic predicates. $N_1 ay$ and $N_2 xuy$ are also quantifier-free and contain only monadic and dyadic predicate symbols. The sets $\{B_0, B_1, B_2 \dots\}$ and $\{B_0', B_1', B_2' \dots\}$ are denoted by $Z(A_0,A_1)$ and $Z'(A_0,A_1)$ respectively. The relation between A_0, A_1, B_i, and B_i' are as follows:

i. $\vdash B_i \supset B_i'$

ii. Every model $\mathfrak{A}'$ of B_i' can be extended to a model $\mathfrak{A}$ of B_i, such that $|\mathfrak{A}| = |\mathfrak{A}'|$.

iii. $i \in A_0 \iff B_i \in N(Z(A_0,A_1))$ (inconsistent)

iv. $i \in A_1 \iff B_i \in F(Z(A_0,A_1))$

v. The class $\{B_i \mid i \notin A_0\}$ is a consistent class.

vi. Let A_4 be a finite subset of A_1. Then the class of formulas $\{B_i \mid i \in A_4\}$ is satisfiable in a finite domain.

vii. Let A_2 be a r.e. set of natural numbers such that $A_2 \cap A_0 = \emptyset$. Then there exists a Büchi model $\mathfrak{A}_2$ for the class $\{B_i \mid i \in A_2\}$ whose diagram is r.e. Moreover, we also have that $i \in A_2 \iff \mathfrak{A}_2(B_i) = T$.

viii. Let A_3 be a recursive set of natural numbers such that $A_3 \cap A_0 = \emptyset$. Then there exists a Büchi model $\mathfrak{A}_3$ for the class A_3 such that the diagram of $\mathfrak{A}_3$ is recursive and such that $i \in A_3 \iff \mathfrak{A}_3(B_i) = T$.

ix. There exists a Krom formula $B'' = B''(A_0,A_1)$ of the form

(21) $$\exists a \forall y N_1'' ay \ \& \ \forall x \exists u \forall y N_2'' xuy$$

such that

$$i \in A_1 \iff \vdash B'' \supset B_i'$$

and

$$i \in A_0 \iff \vdash B'' \supset \neg B_i' .$$

Hence if A_0 and A_1 are recursively inseparable, then B'' is an essentially undecidable theory.

x. If A_0 and A_1 are recursively inseparable then B'' has no recursive Büchi model.

In theorem 3 we have written $B_i = B_i(A_o,A_1)$, $B'_i = B'_i(A_o,A_1)$ and $B'' = B''(A_o,A_1)$ to emphasize that B_i, B'_i and B'' depend on A_o and A_1 . Note also that all B_i contain a fixed number of dyadic predicates, but the number of monadic predicates increases by the order of 2^i in B_i . All B'_i contain a fixed number of predicate symbols.

Sketch of a proof of theorem 3.

In order to define input we use monadic predicates in $M_i xu$. F_o is the only monadic predicate which occurs both in M_i and N_1 . N_2 does not contain monadic predicates. $F_o x$ means x is an input. $N_1 ay$ contains the disjunction $\neg F_o y \vee F_1 ya$. In the formulas B'_i $(i = 0,1,2 \ldots)$ input is defined by $F_o a_j$ where $j = 2^i$. The reason why this will work is that the part $\forall y(N_1 a_o y \ \& \ N_2 a_o a_1 y \ \& \ \ldots \ \& \ N_2 a_{j-1} a_j y \ \& \ F_o a_j)$ of the formula B'_i forces $a_o, a_1, a_2, \ldots, a_j$ to be an intial segment of a Büchi model for $\exists a \forall y N_1 ay \ \& \ \forall x \exists u \forall y N_2 xuy$. Then theorem 3 i, ii, iii, v, vi and viii is easy to prove. In order to prove theorem 3 iv we have to modify the construction somewhat. It is easy to prove that

iv'. $\quad i \in A_1 \implies B_i \in F(Z(A_o,A_1))$

but

iv". $\quad B_i \in F(Z(A_o,A_1)) \implies i \in A_1$

may not be true in general. Suppose that $i \notin A_1$. If $i \in A_o$ then $B_i \notin F(Z(A_o,A_1))$.

Hence suppose that $i \notin A_o$ also. If the registermachine R with input 2^i cycles (goes into a finite loop) we would have that $B_i \in F(Z(A_o,A_1))$. But it is easy to construct the registermachine in such a way that it never cycles. It is still difficult to prove that $B_i \notin F(Z(A_o,A_1))$. We can solve the problem by adding some new binary disjunctions to $N_2 xay$. Let G be a new binary predicate. The new disjunctions are $\neg Gxx \vee \neg Gxx$, $\neg Gyx \vee Gyu$, $\neg F_i xy \vee Gxu$, $\neg F_i yx \vee Gxu$, where $i = 1,2,\ldots,r$. In order to prove theorem 3 iv , we use the fact that the formula

$$\forall x \exists u \forall y ((\neg Gxx \vee \neg Gxx) \& (\neg Gyx \vee Gyu) \& Gxu)$$

is consistent but that it has no finite models.

In order to prove theorem 3 ix and x we shall construct the formula (21) using the same binary predicate letters as in $N_1 ay$ and $N_2 xuy$ in (19) and (20).

Let A be a set of natural numbers, such that $A \cap A_0 = \emptyset$. Then we may define a structure $\mathfrak{A}$ satisfying all formulas B_i and B_i' where $i \in A$, and where $F_j \alpha\beta$ is true $\Longleftrightarrow (\exists i)(i \in A \text{ and } (1,2^i,0) \vdash^* (j,\alpha,\beta))$. The intended model $\mathfrak{A}''$ for (21) is such that

$$F_j \alpha\beta \text{ is true} \Longleftrightarrow (\exists i)(\exists \alpha_1)(\exists \beta_1)((j,\alpha,\beta) \vdash^* (i,\alpha_1,\beta_1) \text{ and } I_i \text{ is } H_1) .$$

Hence in case 1 when I_i is H_1 we add $F_i xy \vee F_i xy$ and for each binary disjunction in table 1 of the form $\neg P \vee Q$ where P and Q are atomic formulas, we also add $P \vee \neg Q$. In this way we obtain $P \equiv Q$, which are what we want, since if $(j_1,\alpha_1,\beta_1) \vdash^* (j_2,\alpha_2,\beta_2)$ then

$$(\exists i)(\exists \alpha_3)(\exists \beta_3)((j_1,\alpha_1,\beta_1) \vdash^* (i,\alpha_3,\beta_3) \text{ and } I_i \text{ is } H_1)$$

$$(\exists i)(\exists \alpha_3)(\exists \beta_3)((j_2,\alpha_2,\beta_2) \vdash^* (i,\alpha_3,\beta_3) \text{ and } I_i \text{ is } H_1)$$

We also add $F_0 y \vee \neg F_1 ya$ to $N_1 ay$ to obtain $N_1'' ay$. These are the main steps in the proof of theorem 3 ix. Suppose that A_0 and A_1 are recursively inseparable. Then the set $\{\alpha \mid F_0 \alpha \text{ is true}\}$ is not a recursive set in a Büchi model for B'' . This proves theorem 3 x. This completes the outline of the proof of theorem 3.

From theorem 3 we obtain the following corollaries.

Corollary 6. The formula

$$\exists a \forall y N_1'' ay \,\&\, \forall x \exists u Hxu \,\&\, \forall x \forall z \forall y (\neg Hxz \vee N_2'' xzy)$$

where N_1'' and N_2'' are as in (21) and H is a new predicate letter, has no recursive model.

Definition 9. A class Z of formulas in first order predicate calculus is called a conservative reduction class iff there exists an effective procedure by which, when an arbitrary formula S in first order predicate logic is given, a corresponding formula S_Z of the class Z can be found such that

i. S is satisfiable $\iff$ S_Z is satisfiable

ii. S is satisfiable in a finite domain $\iff$ S_Z is satisfiable in a finite domain.

From lemma 3 and theorem 3, we obtain the following improvements of theorem 2.

Corollary 7. The classes $\exists\forall \,\&\, \forall\exists\forall \cap Kr$, $\exists\forall\exists\forall \cap Kr$ and $\forall\exists\exists\forall \cap Kr$ are conservative reduction classes.

Theorem 4. The classes $N(\forall\exists\forall \cap Kr)$, $F(\forall\exists\forall \cap Kr)$ and $I(\forall\exists\forall \cap Kr)$ are all non-empty and recursive.

In order to prove theorem 4 we first prove that we reduce the case to consider formulas

(22) $\qquad \forall x \exists u \forall y Mxuy$

which contain atomic parts of the forms

(23) Fxy, Fyx, Fuy, Fyu and

(24) Fxx, Fyy, Fuu ,

only, where F is a binary predicate symbol.

We shall here first consider the set $N(\forall\exists\forall \cap Kr)$. If (22) is satisfiable, then (22) has models whose domain is the integers $\{\ldots -1,0,1,2,\ldots\} = Z$ and such that $M\alpha(\alpha+1)\beta$ is true for every pair of integers α, β .

Let M'xay be the formula obtained by deleting all disjunctions which oontain atomic parts of the form (24). We shall now consider sets of pairs of integers (α_1, β_1) satisfying

(25) $$(\forall\alpha)(\forall\beta)M'a(\alpha+1)\beta \supset (G_1\alpha_o\beta_o \supset G_2(\alpha_o+\alpha_1)(\beta_o+\beta_1))$$

where G_ixy is Fxy, or $\neg$Fxy or Fyx or $\neg$Fyx for some binary predicate Fxy . As in the theory of bounded languages in Ginsburg 1966 we regard Z^n instead of N^n where $N = \{0,1,2,\ldots\}$, to be a subset of the space $\mathcal{R}^n$. Moreover, given subsets C and P of Z^n , let L(C;P) denote the set of all elements in Z^n which can be represented by the form

$$c_o + x_1 + x_2 + \ldots + x_m$$

for some c_o in C and some (possibly empty) sequence $x_1, \ldots, x_m$ of elements of P . C is called the set of constants and P the set of periods of L(C;P) .

$L \subseteq Z^n$ is said to be a <u>linear set</u> if C consists of exactly one element, say $C = \{c\}$, and P is finite, say $\{p_1, \ldots, p_n\}$. A subset of Z^n is said to be <u>semilinear</u> if it is a finite union of linear sets.

Note that we consider here subsets of Z^n instead of N^n as used in the theory for bounded languages.

We can now prove the following lemma.

Lemma 4. The set of pairs (α,β) satisfying (25) is a semilinear set. There exists an effective procedure by which a representation of the semilinear sets can be obtained from $M'xuy$.

Next we prove the following lemma:

Lemma 5. The set of integers α satisfying

(26) $$(\forall\alpha)(\forall\beta)M\alpha(\alpha+1)\beta \supset (H_1\alpha_0 \supset H_2(\alpha_0+\alpha))$$

is a semilinear set. Here H_ix is Fxx or $\neg Fxx$ for some binary predicate F . There exists an effective procedure by which a representation of the semilinear sets can be obtained from $Mxuy$.

The lemmas 4 and 5 are proved almost in the same way as Parikh's theorem which says that if $L \subseteq a^*b^*$ is a contextfree language then the set $\{(i,j) \mid a^ib^j \in L\}$ is semilinear. See Ginsburg 1966, pp. 146-149.

The theorem 4 now follows easily.

References.

Aanderaa, S.O. 1966. A new undecidable problem with application in logic, Doctoral Thesis, Harvard University, Cambridge, Mass., U.S.A.

Büchi, J.R. 1962. Turing Machines and the Entscheidungsproblem. Math. Ann. Vol.148, pp.201-213.

Chang, C.C. and Keisler, H.J. 1962. An improved prenex normal form. J. Symbolic Logic. Vol. 27, pp.317-326.

Dreben, B., Kahr, A.S. and Wang,Hao 1962.Classification of AEA formulas by letter atoms. Bull. Amer. Math. Soc. Vol.68. pp.528-532.

Fischer, P.C., 1966. Turing machines with restricted memory access. Information and Control. Vol.9 pp.364-379.

Ginsburg, S., 1966. The mathematical theory of context-free languages. McGraw-Hill, New York.

Kahr, A.S., Moore, E.F. and Wang, Hao 1962. Entscheidungsproblem reduced to the AEA case, Proc. Nat. Acad. Sci. U.S.A.

Krom, M.R., 1966. A property of sentences that define quasiorder. Notre Dame J. Formal Logic. Vol.7 pp.349-352.

Krom, M.R., 1967a. The decision problem for a class of first order formulas in which all disjunctions are binary. Z. Math. Logik Grundlagen Math. Vol.13 pp. 15-20.

Krom, M.R., 1967b. The decision problem for segregated formulas in first-order logic. Math. Scand. Vol.21 pp.233-240.

Krom, M.R., 1968. Some interpolation theorems for first-order formulas in which all disjunctions are binary. Logique et Analyse. Vol.43. pp.403-412.

Krom, M.R., 1970. The decision problem for formulas in prenex conjunctive normal form with binary disjunctions. J.Symbolic Logic. Vol. 35 pp.210-216.

Maslov, S.Ju., 1964. An inverse method of establishing deducibilities in the classical predicate calculus. Dokl.Akad.Nauk SSSR. Vol 159, pp. 1420-1424.

Minsky, M.L., 1961, Recursive unsolvability of Post's problem of tag and other topics in the theory of Turing machines. Ann. of Math. Vol 74, pp. 437-455.

Minsky, M.L., 1967. Computation: finite and infinite machines, Prentice-Hall.

Rogers, H.Jr. 1967. Theory of recursive functions and effective Computability. McGraw-Hill.

Robinson, A. 1963. Introduction to model theory and to the metamathematics of algebra. North-Holland, Amsterdam.

Shepherdson, J.C., 1965. Machine configuration and word problems of given degree of unsolvability. Z.Math.Logik Grundlagen Math. Vol 11, pp. 149-175.

Shepherdson, J.C. and Sturgies, H.E. 1963. Conputability of recursive functions. J.Assoc.Comput.Mach., Vol 10 pp.217-255.

Shoenfield, J.R., 1967. Mathematical Logic. Addison-Wesley.

Tractenbrot, B.A., 1953. O Rekurajonoj Otdelimosti. Dokl.Akad. Nauk SSSR. Vol 88 pp. 953-955.

Wang, Hao, 1961. Proving theorems by pattern recognition, II. Bell System Tech.J. Vol 40 pp. 1-41.

Wang, Hao. 1962. Domino and the AEA case of the decision problem. Proc.Symposium on Mathematical Theory of Automatic, Polytechnic Institute of Brooklyn, pp. 23-55.